CLAIMS:

1.   A system comprising:

        a set of objects encapsulating computational models; and

        a software program executing within a computer operating environment and having an embedded control module to invoke the computational models in parallel.

2.   The system of claim 1, further comprising a model aggregator to receive input values from the control module and to distribute the input values to the objects.

3.   The system of claim 2, wherein each model includes at least one input and at least one output, and further wherein the model aggregator stores configuration data mapping a set of input slots to the inputs of the models.

4.   The system of claim 3, wherein the configuration data maps a single input slot to multiple inputs of different models.

5.   The system of claim 2, wherein the model aggregator receives predicted output values from the objects and communicates the predicted output values to the control module.

6.   The system of claim 5, wherein the control module displays the predicted output values from the computational models simultaneously.

7.   The system of claim 1, wherein the control module receives input from a user and communicates the inputs to the object models as inputs to the computational models.

8.   The system of claim 1, wherein the software program comprises process management software to manage a manufacturing process.

9.   The system of claim 8, wherein the control module receives measured process data and communicates the measured process data to the object models as inputs to the computational models.

10. The system of claim 2, further comprising a configuration module to select a set of models in response to user input, and to direct the model aggregator to create the set of objects to encapsulate the computational models.

5 11. The system of claim 10, wherein the configuration module, the control module and the set of objects comprises reusable software components conforming to a software component architecture.

12. The system of claim 1, wherein the objects comprise reusable model software
10 components arranged as one or more dynamic linked libraries (DLLs) invoked by the control module.

13. The system of claim 1, further comprising one or more dynamic linked libraries (DLLs) that implement:
   the set of objects;
15   the control module;
   a configuration module invoked by the software program to configure the set of objects in response to user input; and
   a model aggregator to receive input values and commands from the control module
20 and to distribute the input values and commands to the objects for invoking the computational models.

14. A system comprising:
   a set of objects having generic interfaces for controlling encapsulated
25 computational models;
   process management software executing within a computer operating environment to control a manufacturing process, wherein the process management software includes an embedded control module;
   a model aggregator to receive input values and commands from the control module
30 and to distribute the input values and commands to the objects via the generic interfaces, wherein the control module directs the model aggregator to invoke the computational models in parallel.

14

15.   The system of claim 14, further comprising configuration data mapping a set of N input slots to M inputs of the models, wherein M is greater than or equal to N.

16.   The system of claim 14, wherein the control module simultaneously displays:

(i) predicted output values received from the model aggregator and generated by the computational models; and

(ii) process data measured from the manufacturing process.

17.   The system of claim 14, further comprising a configuration module to select the computational models in response to user input, and to direct the model aggregator to create the set of objects to encapsulate the computational models.

18.   The system of claim 17, wherein the configuration module, the control module and the set of objects comprises reusable software components.

19.   A computer-readable medium comprising instructions causing a processor to:

instantiate a set of objects encapsulating computational models and including generic interfaces for invoking the computational models;

instantiate a model aggregator to distribute input values to the objects and to receive predicted output values from the objects; and

instantiate a control module to receive the output values from the model aggregator and to display the output values.

20.   The computer-readable medium of claim 19, wherein the objects, the model aggregator and the control module comprise reusable components, and further wherein the control module is arranged for embedding within an executable program.

21.   The computer-readable medium of claim 20, wherein the objects, the model aggregator and the control module are arranged as dynamic linked libraries (DLLs).

22.   The computer-readable medium of claim 19, wherein the instructions further comprise a configuration module to select the computational models in response to user input, and to direct the model aggregator to create the objects encapsulating the models.

23.   The computer-readable medium of claim 19, wherein the instructions further comprise an executable software program embedding the control module.

24.   The computer-readable medium of claim 23, wherein the executable software program comprises process management software to manage a manufacturing process.

25.   A method comprising:
       encapsulating a set of computational models within objects, wherein each object provides a generic interface for invoking the encapsulated computational model;
       embedding a control module within an executable software program; and
       invoking the set of objects from the control module to execute the computational models in parallel.

26.   The method of claim 25, further comprising:
       defining a set of input slots; and
       mapping inputs of the encapsulated models to the input slots.

27.   The method of claim 26, further comprising mapping at least two of the inputs of the models to a common input slot.

28.   The method of claim 26, further comprising:
       receiving input values from the control module, and
       distributing the input values to the objects according to the mapping.

29.   The method of claim 25, further comprising:
       selecting the models in response to user input; and
       creating one of the objects for each of the selected models.

30.   The method of claim 25, wherein invoking the set of objects comprises:
       querying a model aggregator to identify the objects; and
       directing the model aggregator to provide input values to and receive predictive output values from the interfaces of the objects.

31.   The method of claim 25, further comprising:

        receiving measured data from a manufacturing process; and

        communicating the measured data to the objects for use during model execution.

5   32.   The method of claim 25, further comprising:

        receiving an input value from a user; and

        directing the input value to the appropriate objects for use during model execution.

     33.   The method of claim 25, further comprising:

10   receiving predictive output values from the objects; and

        presenting the predictive output values to a user.

     34.   The method of claim 33, further comprising presenting the output values

simultaneously.